AFRL-IF-RS-TR-2004-25
Final Technical Report
February 2004

MAPLE: MULTI-AGENT PLANNING, LEARNING, AND EXECUTION

Carnegie Mellon University

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. J388

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

**AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK**

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2004-25 has been reviewed and is approved for publication.

APPROVED: /s/
FRANK H. BORN
Project Engineer

FOR THE DIRECTOR: /s/
JAMES A. COLLINS, Acting Chief
Information Technology Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
FEBRUARY 2004

3. REPORT TYPE AND DATES COVERED
FINAL **Apr 98 – Apr 03**

4. TITLE AND SUBTITLE

MAPLE: MULTI-AGENT PLANNING, LEARNING, AND EXECUTION

5. FUNDING NUMBERS
G - F30602-98-2-0137
PE - 63760E
PR - AGEN
TA - T0
WU - 04

6. AUTHOR(S)

Sebastian Thrun

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington VA 22203-1714

AFRL/IFTB
525 Brooks Road
Rome NY 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

AFRL-IF-RS-TR-2004-25

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Frank H. Born/IFTB/(315) 330-4726 Frank.Born@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*

The fundamental problems addressed in this report pertained to the coordination of multi-agent software systems acting in dynamic, physical environments. The Multi-Agent Planning MAPLE project focused on the development and evaluation of new algorithms for integrating information and for coordinating the actions of large multi-agent teams. Three different aspects of multi-agent systems are in this report. First, this project developed a prototype of a secure decentralized database, which made it possible to disperse information among multiple agents in a secure and scalable way. Second, the project developed a multi-agent system for acquiring and integrating information in large dynamic environments. Third was a planning and multi-agent coordination technique for enabling large teams of autonomous agents to perform missions in highly dynamic and partially observable environments.

The resulting algorithms were implemented and evaluated predominately using a simulator called “MapleSim”, which simulated the country of Honduras after the devastation caused by Hurricane Mitch in 1998.

14. SUBJECT TERMS
Agent Based Software, Multi-agent system, planning, scheduling, coordination, autonomous software

15. NUMBER OF PAGES 19

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Table of Contents

1 Executive Summary of Research and Findings

The report summarizes the achievements and accomplishments of the project MAPLE (Multi-Agent Planning, Learning, and Execution), sponsored by DARPA under contract F30602-98-2-0137. Overall, the work carried out under this contract has led to 36 publications and a software prototype connected to the Control of Agent Based Systems (CoABS) Grid.

The fundamental problem addressed in this project pertained to the coordination of multi-agent system acting in dynamic, physical environments. An example of such a setting is a set of unmanned ground vehicles, which cooperatively engage in humanitarian missions in the aftermath of a natural disaster (e.g., a hurricane). The MAPLE project focused on the development and evaluation of new algorithms for integrating information and for coordinating the actions of large multi-agent teams.

Our scientific effort focused on three different aspects of multi-agent systems, which are outlined in the detailed approach section of this report. First, under the initial project name "BORG," we developed a prototype of a secure decentralized database, which made it possible to disperse information among multiple agents in a secure and scalable way. Second, we developed a multi-agent system for acquiring and integrating information in large dynamic environments. Third, we developed planning and multi-agent coordination techniques for enabling large teams of autonomous agents to perform missions in highly dynamic and partially observable environments. While our initial research focused on the development of the BORG system, the majority of resources were spent developing algorithms for the last two items: techniques for multi-agent information integration, and techniques for coordinating many physical agents.

The resulting algorithms were implemented and evaluated predominately in the context of the Mixed Initiative Agent Team Administration (MIATA) Technology Integration Experiment (TIE), which was carried out jointly by a number of CoABS contractors. We developed the backbone software system for this TIE, a simulator called "MapleSim," which simulated the country of Honduras after the devastation caused by Hurricane Mitch in 1998. MapleSim has been connected to the CoABS Grid, and a number of contractors controlled air and ground operations through the infrastructure provided by the Grid and MapleSim. A second contribution to this demonstration was the control of trucks on the ground. This problem was challenging in that the conditions on the ground were largely unknown; furthermore, the environment was highly dynamic.

Other evaluations were carried out in the context of multi-robot systems. We have expanded this research to embrace the coordination of multiple ground vehicles when exploring unknown physical environments, and the coordination of multiple unmanned air vehicles when acquiring information in dynamic battlefield situations (e.g., finding SAM sites). The results of this project have successfully been transitioned into related DARPA projects, such as the DARPA TMR project and the DARPA MICA project.

2 Main Accomplishments

The main accomplishments of the MAPLE project are as follows:

- **Acc-1:** We developed a prototype of a secure scalable database, known as BORG. Through this preliminary prototype, we have demonstrated scalability and security for this decentralized database.
- **Acc-2:** We developed scalable algorithms for multi-agent planning and coordination in dynamic and partially observable environments. In particular, we developed:
  - **Acc-2a:** Partially observable Markov decision process techniques for coordinating many agents in the process of information gathering [15, 25].
  - **Acc-2b:** Scalable greedy algorithms for coordinating multi-agent systems when exploring unknown environments and acquiring detailed models thereof [9, 27].
  - **Acc-2c:** Auction algorithms for optimal coordination of multiple agents operating in adversarial environments [6].
  - **Acc-2d:** Motion planning algorithms for deconflicting large number of moving entities (e.g., ground vehicles) that share the same environment, each pursuing different goals [4, 5].
- **Acc-3:** We developed scalable multi-agent information integration algorithms.
  - **Acc-3a:** We developed a number of algorithms for exploring and mapping unknown environments with teams of mobile agents [18, 19, 20, 16, 33, 34, 28]. The paper [28] won the best conference paper award at the 2000 ICRA Conference (selected as the sole winner from over 1,100 submissions).
  - **Acc-3b:** We developed a number of specialized algorithms for exploring and acquiring models of partially known - partially unknown environments, such as the country Honduras after the devastation caused by Hurricane Mitch [1, 10]. Our algorithms specifically address dynamic environments, in which aspects may change over time as the data is acquired [10, 36].
  - **Acc-3c:** We developed techniques for tracking moving entities in dynamic environments with teams of mobile robots [23, 22]. The paper [23] won the best Student Conference paper award at AAMAS, the international multi-agent conference.
  - **Acc-3d:** We developed a number of advanced probabilistic tracking techniques for single-agent systems [30, 35].
- **Acc-4:** We developed MapleSim, a discrete event simulator of the country Honduras after the Hurricane Mitch devastated large parts of the country [15].

- **Acc-5:** We interfaced MapleSim and our multi-agent coordination system to the CoABS Grid, and provided full support to various DARPA demonstrations and TIEs.

- **Acc-6:** Finally, we developed a visualization tool for the CoABS grid which was used in a number of DARPA demonstrations and experiments throughout the CoABS program.

3 Publications

The following publications directly arose out of the MAPLE project and were fully or partially funded under the contact F30602-98-2-0137. The DARPA support has been acknowledged.

[1] D. Avots, E. Lim, R. Thibaux, and S. Thrun. A probabilistic technique for simultaneous localization and door state estimation with mobile robots in dynamic environments. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland, 2002.

[2] M. Bennewitz, W. Burgard, and S. Thrun. Constraint-based optimization of priority schemes for decoupled path planning techniques. In *Proceedings of the Joint German Austrian Conference on Artificial Intelligence*. Springer Verlag, 2001.

[3] M. Bennewitz, W. Burgard, and S. Thrun. Optimizing priority schemes for decoupled path planning techniques. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS'2001)*, Hawaii, 2001.

[4] M. Bennewitz, W. Burgard, and S. Thrun. Optimizing schedules for prioritized path planning of multi-robot systems. In *IEEE International Conference on Robotics and Automation (ICRA)*, Seoul, Korea, 2001. ICRA.

[5] M. Bennewitz, W. Burgard, and S. Thrun. Adapting navigation strategies using motions patterns of people. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[6] C. Bererton, G. Gordon, and S. Thrun. Auction mechanism design for multi-robot coordination. Submitted for publication, 2003.

[7] M. Berna, B. Lisien, B. Sellner, G. Gordon, F. Pfenning, and S. Thrun. A learning algorithm for localizing people based on wireless signal strength that uses labeled and unlabeled data. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003. IJCAI.

[8] R. Biswas, B. Limketkai, S. Sanner, and S. Thrun. Towards object mapping in dynamic environments with mobile robots. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland, 2002.

[9] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, 2000. IEEE.

[10] W. Burgard, D. Hähnel, R. Triebel, and S. Thrun. Mapping with mobile robots in dynamic environments. In *Proceedings of the Snowbird Workshop "Machines That Learn"*, Snowbird, UT, 2003. NIPS Foundation. Extended abstract.

[11] D. Hähnel, W. Burgard, B. Wegbreit, and S. Thrun. Towards lazy data association in SLAM. In *Proceedings of the 11th International Symposium of Robotics Research (ISRR'03)*, Sienna, Italy, 2003. Springer.

[12] D. Hähnel, D. Fox, W. Burgard, and S. Thrun. A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, 2003.

[13] D. Hähnel, R. Triebel, W. Burgard, and S. Thrun. Map building with mobile robots in dynamic environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[14] B. Limketkai, R. Biswas, and S. Thrun. Learning occupancy grids of non-stationary objects with mobile robots. In *Proceedings of the International Symposium on Experimental Robotics (ISER)*, Sant'Angelo d'Ischia, Italy, 2002.

[15] L. Lita, J. Schulte, and S. Thrun. A multi-agent system for agent coordination in uncertain environments (short paper). In *Proceedings of Agents*, Montreal, CA, 2001.

[16] Y. Liu and S. Thrun. Results for outdoor-SLAM using sparse extended information filters. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[17] M. Montemerlo and S. Thrun. Simultaneous localization and mapping with unknown data association using FastSLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[18] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Edmonton, Canada, 2002. AAAI.

[19] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003. IJCAI.

[20] E. Nettleton, S. Thrun, and H. Durrant-Whyte. Decentralised slam with low-bandwidth communication for teams of airborne vehicles. In *Proceedings of the International Conference on Field and Service Robotics*, Lake Yamanaka, Japan, 2003.

[21] J. Nieto, J. Guivant, E. Nebot, and S. Thrun. Real time data association for FastSLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[22] M. Rosencrantz, G. Gordon, and S. Thrun. Decentralized sensor fusion with distributed particle filters. In *Proceedings of the Conference on Uncertainty in AI (UAI)*, Acapulco, Mexico, 2003.

[23] M. Rosencrantz, G. Gordon, and S. Thrun. Locating moving entities in dynamic indoor environments with teams of mobile robots. In *Proceedings of Autonomous Agents and Multi-Agent Systems*, Melbourne, Australia, 2003.

[24] N. Roy and S. Thrun. Motion planning through policy search. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland, 2002.

[25] J. Schulte and S. Thrun. A heuristic search algorithm for acting optimally in markov decision processes with deterministic hidden state. Internal Report, 2001.

[26] R. Simmons, D. Apfelbaum, W. Burgard, M. Fox, D. an Moors, S. Thrun, and H. Younes. Coordination for multi-robot exploration and mapping. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Austin, TX, 2000. AAAI.

[27] R. Simmons, D. Apfelbaum, D. Fox, R.P. Goldmann, K.Z. Haigh, D.J. Musliner, M. Pelican, and S. Thrun. Coordinated deployment of multiple heterogeneous robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2000.

[28] S. Thrun. Towards programming tools for robots that integrate probabilistic computation and learning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, 2000. IEEE.

[29] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, 20(5):335–363, 2001.

[30] S. Thrun, Langford. J., and V. Verma. Risk sensitive particle filters. In *Advances in Neural Information Processing Systems 14*. MIT Press, 2002.

[31] S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte, and A.Y. Ng. Simultaneous mapping and localization with sparse extended information filters. In J.-D. Boissonnat, J. Burdick, K. Goldberg, and S. Hutchinson, editors, *Proceedings of the Fifth International Workshop on Algorithmic Foundations of Robotics*, Nice, France, 2002.

[32] S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte, and A.Y. Ng. Simultaneous mapping and localization with sparse extended information filters:theory and initial results. Technical Report CMU-CS-02-112, Carnegie Mellon University, Computer Science Department, Pittsburgh, PA, 2002.

[33] S. Thrun and Y. Liu. Multi-robot SLAM with sparse extended information filers. In *Proceedings of the 11th International Symposium of Robotics Research (ISRR'03)*, Sienna, Italy, 2003. Springer.

[34] S. Thrun, Y. Liu, D. Koller, A.Y. Ng, Z. Ghahramani, and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. Submitted for journal publication, April 2003.

[35] V. Verma, R. Simmons, and S. Thrun. Variable resolution particle filter. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003. IJCAI.

[36] C.-C. Wang, C. Thorpe, and S. Thrun. Online simultaneous localization and mapping with detection and tracking of moving objects: Theory and results from a ground vehicle in crowded urban areas. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

4 Detailed Project Report

4.1 BORG: Secure Scalable Database

In the initial 12 Project months, the MAPLE project focused on the development of a secure and scalable database system. The idea of the BORG prototype is as follows: Imagine you want to secure a piece of information against a possible enemy. Then cut this piece of information into pieces and duplicate these pieces among multiple agents. The key insight of the BORG prototype is that this dispersion can be done such that an opponent has to compromise at least M out of N agents to recover even a single bit of the protected information, for arbitrary $M \leq N$. Compromising $M - 1$ is not sufficient; in fact, no information can be recovered whatsoever by compromising fewer than M agents (not even partial information). By making M a large number, the system becomes virtually impenetrable. It becomes highly secure against information warfare attacks.

In the initial twelve project months, we developed a prototype of this system. Issues addressed during this period pertained to the manipulation of information (e.g., adding two values, executing simple programs) without jeopardizing the information itself. The work on BORG was discontinued based on feedback by the PM (Jim Hendler), who changed the focus of the MAPLE project to real-world planning and multi-agent coordination, and the CoABS Grid.

4.2 Multi-Agent Planning And Coordination

The core of the MAPLE project focused on the development of multi-agent planning and coordination techniques, for coordinating large numbers of physical agents operating in highly unpredictable and uncertain environments. The fact that those agents act in physical spaces leads to constraints in which the physical whereabouts play a major role in agent coordination. Further, the fact that the environment is partially unknown and dynamic mandates representations that can account for the inherent uncertainty in such environment. As a result, the agents must incorporate knowledge of their own ignorance into decision making, and not just base the decision on their best guess of the current state of the world (as is common practice). Our approach, thus, differs fundamentally from previous work on multi-agent coordination, which has inadequately addressed the issue of uncertainty and lack of complete world knowledge. All our algorithms are built on this insight, and all of them address the inherent uncertainty in the word.

4.2.1 POMDP Techniques For Information Gathering With Many Agents

A first set of algorithms described in [15, 25] relies on a basic insight known as partially observable Markov decision process, or POMDP. POMDPs are best explained as follows: Since the world is stochastic, it is best modeled as Markov process, which is a stochastic chain where the state of the world at time t depends stochastically on its state at time $t - 1$.

However, not all of the state is observable. Instead the agents have sensors that provide them with incomplete and noisy projections of the (unobservable) state. Markov chains with partially observable state spaces are known as partially observable Markov chains. Lastly, the agents have to make control decisions (e.g., where to move), hence the term “decision process” in POMDPs.

Our first set of techniques, which formed the core of our approach in the MIATA TIE, is based on the assumption that the unknown state is discrete: For example, roads in Honduras might be passable or not after Hurricane Mitch destroyed much of the county’s infrastructure; bridges might have collapsed. The challenge then is to reach a number of goal locations (e.g., population centers) under this initial uncertainty about the situation in the POMDPs. In [25], we developed efficient search algorithms for finding optimal exploration-exploitation plans, which not only provided a single best route to a set of goals, but also a full set of contingencies on what to do should a path be blocked. On top of this, we developed a multi-agent arbitration scheme for assigning individual agents to individual goals. This algorithm evaluated the contingency plans provided by the individual agents, and assigned agents to goals based on the expected time of arrival (in fact, based on the entire probabilistic profile as to when an agent could be assumed to arrive at a goal). In [15], we increased the efficiency of this approach by several orders of magnitude through the introduction of abstraction, in which entire subparts of a large POMDP were solved first and represented by a single complex arc in an abstract model of the world. All of these approach were brought to bear in the MIATA TIE, and they were thoroughly evaluated [25].

4.2.2 Coordinating Multi-Agent Exploration

In [9, 27], we further developed multi-agent exploration techniques which we subsequently implemented on physical robots. The following image shows one of our robot systems (Figure 1):

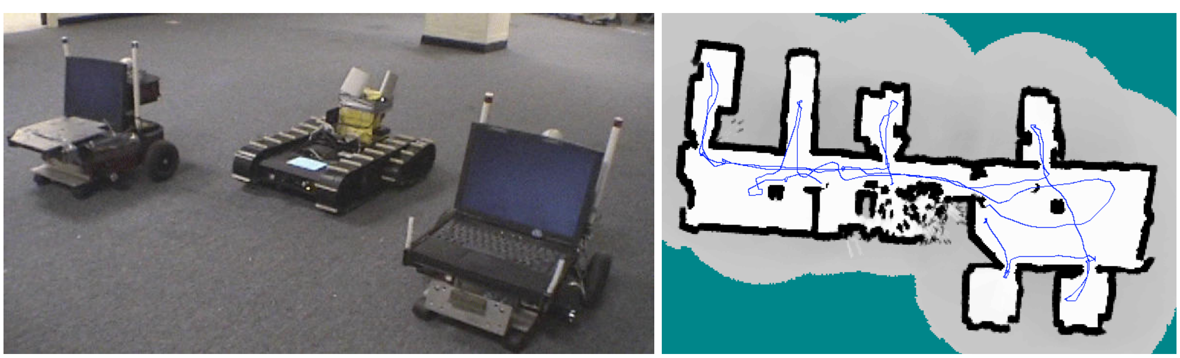

Figure 1: Robots (left) and map with exploration path (right).

The problem addressed in this research is a restricted version of the POMDP above. In particular, the environment is static and the robot can greedily explore it. However, what

makes this case difficult is the extremely large number of unknowns: The environment is represented by a grid consisting of tens of thousands of grid cells. Here the agents have to rely in their sensors to infer the complex structure of the environment as they go. The sensor interpretation and data fusion has to take place in real-time.

We have developed scalable techniques for doing all this. In particular, our approach coordinates multiple robots during exploration and mapping. It interprets sensor data and enables the robots to acquire a single, combined map. As described in detail in [9, 27], our approach involves an auction-style algorithm in which agents "bid" for individual places to explore. A high-level agent then assigns agents to places. This assignment is constantly revised to accommodate the ever-changing state of knowledge of the robot system. Empirical comparisons in [9, 27] show that our approach is superior to uncoordinated exploration techniques, commonly found in previous literature.

4.2.3 Auction Algorithms

Carrying this work a step further, we have recently developed auction-style multi-agent coordination algorithms that are more powerful [6]. In particular, in past work by us and others (including the work above), the coordination was based on a fixed set of prices for shared commodities (e.g., a fixed reward for exploring an unknown location). However, in cases where multi-agent systems share complex commodities (e.g., multiple trucks in Honduras have to share a limited supply of fuel), the costs for using such resources in auction-style algorithms is usually not known. In fact, one can argue that setting appropriate action costs amounts to solving the harder part of the problem. Once such costs have been determined, the problem is easily decomposed into single-agent planning and execution problems.

Our approach in [6] gets at the core of this issue: It alternates a conventional individual-agent planning phase with a phase in which an auctioneer determines the optimal costs to assign to individual commodities and constraints that arise between multiple agents. In this way, a much broader class of problems is made amenable to multi-agent auction-style algorithms, and the overall performance of the system is improved. We have demonstrated this in the contact of a number of simulated multi-agent problems described in [6].

4.2.4 Motion Planning for Large Numbers of Ground Vehicles

In [4, 5], we have specifically addressed the problem of multiple physical agents sharing the same physical environment. Here we model ground vehicles that have to deconflict their motion through the environments. Such problems are well-known to all of us; Just consider the problem of tens of thousands of people commuting to work through a limited network of streets and intersections.

We have developed a randomized set of algorithms that improves over the previous state of the art. Our approach uses prioritized planning techniques to develop an initial multi-agent plan, which is subsequently refined by improving the order at which individual vehicles

can determine their local routes. The approach was tested in a complex network of roads and intersections reminiscent (but not identical) to the scenario used in the MIATA TIE. We have shown that our approach is significantly more successful in deconflicting vehicle motion than classical multi-agent path planning algorithms. However, our approach assumes a deterministic world, which makes it less powerful than the POMDP algorithms described above (in fact, it really addresses a different problem). Detailed descriptions of the results can be found in [4, 5].

4.3 Multi-Agent Information Integration

In addition to multi-agent planning and coordination algorithms, we developed a suite of scalable multi-agent information integration algorithm. The problem underlying all this work is easily formulated: If N agents operate in a dynamic and uncertain world, how can they efficiently integrate their local observations into a single, consistent world knowledge. One obvious solution would be to design a central information agent (a "master agent") with which all agents communicate. However, such a system would be characterized by a single point of failure; moreover, it would be subject to scaling limitations due to the communication overhead involved for the master agent. Our work is strictly decentralized: Communication occurs only between neighboring agents, and the global model emerges through local communication. The resulting algorithms scale to any number of agents and are highly reliable against failure of individual network components. We view this work as an important step towards the vision of network-centric computing. Further, this work is a cornerstone of the multi-agent systems developed in the MAPLE program.

4.3.1 Mapping Unknown Environments

In a number of publications [18, 19, 20, 16, 33, 34, 28], we have developed algorithms for information integration in the context of map acquisition of unknown environments. This problem is one of the hardest in robot perception: It involves vehicles that measure nearby entities as they move about their environments, seeking to model it in a consistent manner. It has an important limiting assumption, namely that the environment is static. This assumption is common in the literature on mapping; below we will discuss our approaches for dynamic environments.

Possibly the most important result of the MAPLE project, documented in [20, 16, 31, 33, 34], is a technique for information integration that is truly scalable to any number of agents. The idea is to represent the model probabilistically, by the posterior over all maps given the measured data. Further, these approaches represent this probabilistic estimate in its information form, which technically amounts to a log-likelihood. Information is additive. Nearby agents, thus, can add their information, and pass the sum on to other neighbors. This large network of information-adding entities can accommodate any number of agents.

Our work has specifically focused on keeping the message sizes small even if the poste-

rior is complex. In problems where the sensors move (e.g., trucks move through Honduras), the exact updates grow over time. This is because uncertainty in vehicle motion can affect increasingly large fractions of the map. We have developed approximate algorithms that communicate the gist of new information immediately, but delay the communication pertaining to remote locations. In doing so, the message size is kept small; in fact, its size is independent of the size of the world and the total sensor history. However, one of the key questions is whether the resulting posterior is still consistent with the one that would be obtained by costly communication algorithms with large message sizes. We have shown—theoretically and empirically—that this is still the case. Further, we have achieved excellent empirical results when compared to the non-approximate (but costly) solution of communicating all sensor data back to a master agent. The papers [34, 31] develop this approach for a single agent system. Work in [33, 16] extends this approach to multi-agent system, under various flavors of uncertain knowledge. The paper [20] describes our efficient communication technique. This work was carried out in cooperation with researchers at the University of Sydney, and since has been implemented for teams of unmapped air vehicles.

4.3.2 Modeling Dynamic Environments

In [1, 10, 36], we have developed a number of algorithms particularly suited for modeling dynamic environments. The key idea here is that some of the environment is static (e.g., the layout of a place), whereas other quantities can change over time. This work overcomes a critical assumption made in the information integration work described thus far, in that it allows for non-static aspects of the environment to exist.

The work in [1] applies a so-called Rao-Blackwellized particle filter to the problem of modeling the state of certain dynamic objects. It is cast as the problem of learning the status of doors in an indoor environment with a mobile robot, but it is equally applicable to the problem of learning the state of bridges and roads in the MIATA TIE. More relevant is the work in [10, 36], which addresses the problem of acquiring maps of environments with static and dynamic aspects. The paper [10] applies the expectation maximization algorithm (EM) to separate static from dynamic aspects, and then simply applies a conventional mapping algorithm for modeling the static aspects of the environment. In [36], we model both, static and dynamic aspects. The technique detects and tracks moving objects in the environment, and makes those track part of the overall model. We have demonstrated all three of those algorithms in the context of physical robot implementations, using indoor and outdoor robots. In our experiments, we found that models of unprecedented complexity could be acquired—in fact, we know of no other work that would address modeling issues in dynamic environments of similar complexity.

4.3.3 Tracking Moving Entities With Many Agents

In [23, 22], we address the problem of tracking moving objects under massive occlusion, with large numbers of moving agents. The motivating scenario was one by which an intruder seeks to move through an environment undetected. Mobile robots have to move through the environment in ways that guarantee that such an intruder be found. This is yet another example of information integration in a dynamic environment. Static information integration techniques are inapplicable.

Our approach relies on a carefully designed communication protocol between neighboring agents. Agents analyze their local beliefs. They query neighbors with hypotheses (e.g., intruder took path X) that they deem to be likely given their observations. Neighbors then respond with confirming or contradicting evidence is such evidence is available; otherwise they communicate the query to their own neighbors. The resulting algorithm was found to be highly efficient in integrating information in such dynamic environments. Physical tests with 4 robots were complimented with simulations of up to 100 robots. We found that the communication overhead could be cut down by several orders of magnitude when compared to a centralized system, where each agent communicates all observations to a central high-level agent. Further, this algorithm is a first scalable algorithm to tracking moving entities with vast amounts of occlusion, with virtually any number of agents.

4.3.4 Advanced Probabilistic Tracking Techniques For Single-Agent Systems

Finally, we have developed a number of algorithms for tracking belief states in POMDPs that advance the state of the art in specific directions [30, 35]. In [30], we developed a particle filter tracking algorithm that considers the performance task at hand: Instead of just maximizing the accuracy of the tracked belief state, it does so relative to the expected cost of tracking errors. The result is quite different from traditional tracking algorithms: By taking the costs into account, the computational resources are mostly focused to parts of the space where failures and errors matter the most. We are not aware of previous work that would fold costs into the basic tracking system.

In [35], we further extended this idea to incorporate hierarchical representations. The problem addressed here is tracking in the context of complex diagnostic systems, in which a system can have a huge number of possible failure states. Tracking all of them is prohibitively expensive. The approach in [35] relies on abstraction to track groups of possible failures, and refines those tracks dynamically as additional information arises. This approach has successfully been applied to online diagnostics of the Hyperion robot, a robot recently deployed in a desert in Chile.

4.4 MapleSim: A Simulator of Ground and Air Vehicles in Honduras

The MAPLE team played a major role in the support of the MIATA TIE. In addition to the algorithms described above, we developed a discrete event simulation platform known as MapleSim [15]. A view of this system is shown in Figure 2:

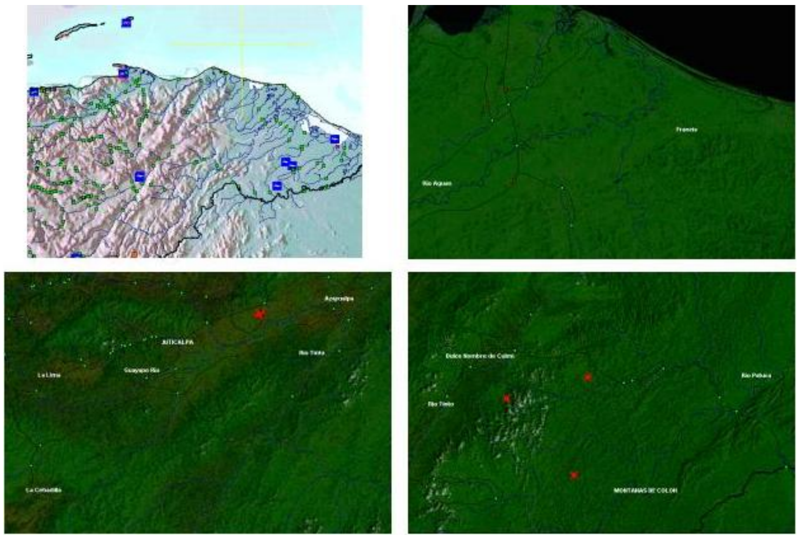

Figure 2: Screenshots of the MapleSim simulation environment (Honduras).

In essence, this simulator can model any geographic region based on data widely available trough the USGS and other services. The simulator models air and ground vehicles. It also models storms and their effect on the environment's infrastructure. In particular, it models destruction of navigation infrastructure and humanitarian requirements arising out of problems such as ground water contamination. The MapleSim system formed the backbone of the MIATA TIE, in which it was used to model the situation in Honduras after Hurricane Mitch destroyed up to 90% of the infrastructure. It has, however, been used for a number of other geographic regions, by a number of contractors in the DARPA CoABS program.

The development and support of MapleSim consumed about a third of the overall resources provided under this contract. The simulation platform consists of a core, which can be run under the Windows or Linux operating system. The core connects to a graphical visualization tool called "SimSpy," which makes it possible to visualize the situation and affect it (e.g., by manually destroying bridges in Honduras). The system also possesses a full API for controlling individual vehicles, and receiving sensor data collected by these vehicles. The types of vehicles modeled in our approach include ground vehicles (trucks), fixed wing aircraft and rotorcrafts. The simulator can model up to 100 such vehicles at a time, at user-selected time scales. In addition, MapleSim offers a limited model of natural disaster effects, and the resulting damage to the road structure. This facility sets MapleSim aside from a number of alternative simulation tools, which tend to model the environment as more static (and there-

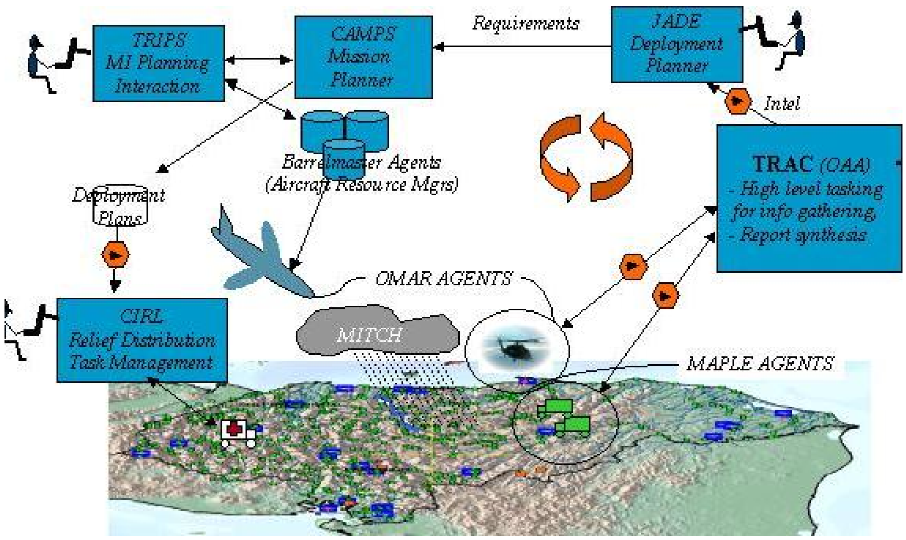

Figure 3: Architecture of the MIATA TIE.

fore more predictable). The MapleSim software has been made available via our project Web site `http://www.cs.cmu.edu/~maple`.

4.5 Porting and Interfacing with CoABS Grid

Major effort was spent on porting software to the CoABS Grid. We developed a C/C++ interface that formed the core of another contractor's work on porting software to the Grid. Based on this interface, we made a number of services available for use by others: the MapleSim simulator, and our agent-based control system for coordinating large number of ground vehicles. The system played a major role in the MIATA TIE, in which multiple contractors cooperatively studied the use of the Grid and agent-based technology in highly dynamic and unpredictable environments. The porting efforts, however, were not fully completed for our most recent software developments.

4.6 CoABS Grid Visualization Tools

Finally, we developed a graphical visualization tool to monitor and analyze the message flow in the Grid. This tool applied a number of graph visualization algorithms to visualize the complex and dynamic network of interacting entities in the CoABS Grid. This tool was used in a number of TIEs throughout the CoABS Program.